



STACK IN DS  
C PROGRAM TO IMPLEMENT  
STACK

# Exception Handling in Java

The **Exception Handling in Java** is one of the powerful *mechanism to handle the runtime errors* so that the normal flow of the application can be maintained.

In this tutorial, we will learn about Java exceptions, it's types, and the difference between checked and unchecked exceptions.

## What is Exception in Java?

**Dictionary Meaning:** Exception is an abnormal condition.

In Java, an exception is an event that disrupts the normal flow of the program. It is an object which is thrown at runtime.

## What is Exception Handling?

Exception Handling is a mechanism to handle runtime errors such as `ClassNotFoundException`, `IOException`, `SQLException`, `RemoteException`, etc.



### Advantage of Exception Handling

The core advantage of exception handling is **to maintain the normal** normally disrupts the normal flow of the application; that is why we need scenario:

```
statement 1;  
statement 2;  
statement 3;  
statement 4;  
statement 5;//exception occurs  
statement 6;  
statement 7;  
statement 8;  
statement 9;  
statement 10;
```

Suppose there are 10 statements in a Java program and an exception occurs at statement 5; the rest of the code will not be executed, i.e., statements 6 to 10 will not be executed. However, when we perform exception handling, the rest of the statements will be executed. That is why we use exception handling in [Java](#).

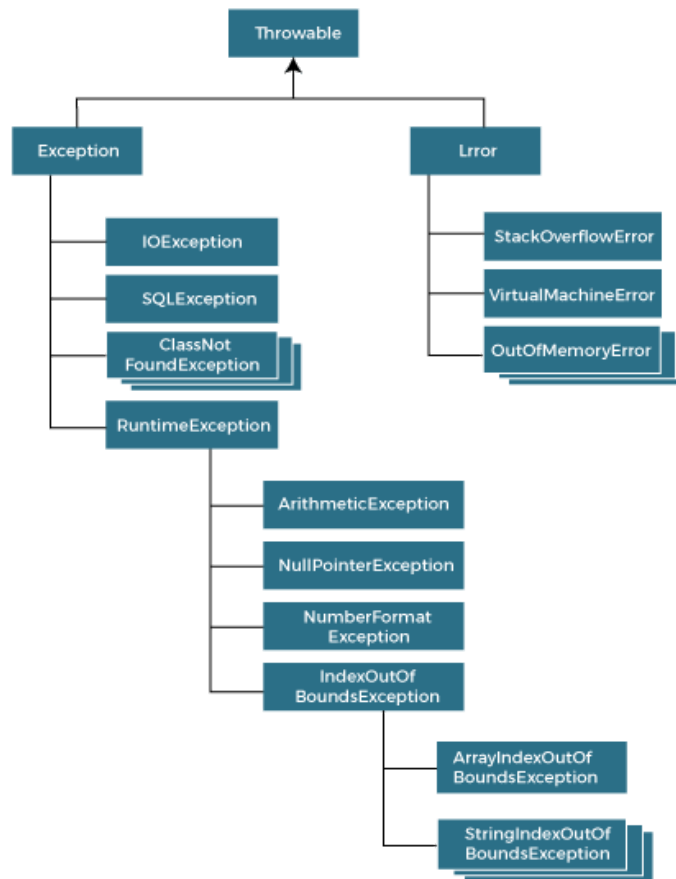
### *Do You Know?*

- What is the difference between checked and unchecked exceptions?
- What happens behind the code `int data=50/0;`?
- Why use multiple catch block?
- Is there any possibility when the finally block is not executed?
- What is exception propagation?
- What is the difference between the throw and throws keyword?
- What are the 4 rules for using exception handling with method overriding?

## Hierarchy of Java Exception classes

The `java.lang.Throwable` class is the root class of Java Exception hierarchy inherited by two subclasses: `Exception` and `Error`. The hierarchy of Java Exception classes is given below:

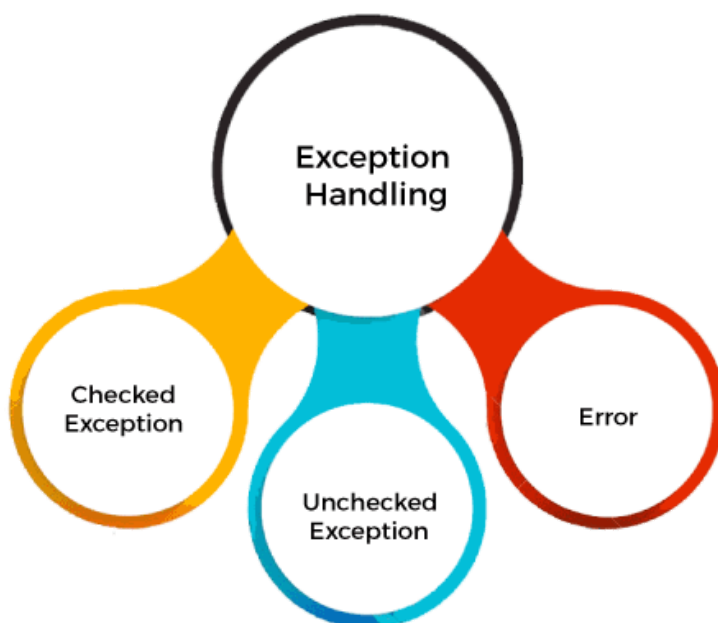




## Types of Java Exceptions

There are mainly two types of exceptions: checked and unchecked. An error is considered as the unchecked exception. However, according to Oracle, there are three types of exceptions namely:

1. Checked Exception
2. Unchecked Exception
3. Error



## Difference between Checked and Unchecked Exceptions

### 1) Checked Exception

The classes that directly inherit the Throwable class except RuntimeException and Error are known as checked exceptions. For example, IOException, SQLException, etc. Checked exceptions are checked at compile-time.

### 2) Unchecked Exception


The classes that inherit the RuntimeException are known as unchecked exceptions. For example, ArithmeticException, NullPointerException, ArrayIndexOutOfBoundsException, etc. Unchecked exceptions are not checked at compile-time, but they are checked at runtime.

### 3) Error

Error is irrecoverable. Some example of errors are OutOfMemoryError, VirtualMachineError, AssertionError etc.

## Java Exception Keywords

Java provides five keywords that are used to handle the exception. The following table describes each.

Keyword	Description
try	The "try" keyword is used to specify a block where we should place an exception code. It means we can't use try block alone. The try block must be followed by either catch or finally.
catch	The "catch" block is used to handle the exception. It must be preceded by try block which means we can't use catch block alone. It can be followed by finally block later.
finally	The "finally" block is used to execute the necessary code of the program. It is executed whether an exception is handled or not.
throw	The "throw" keyword is used to throw an exception.
throws	The "throws" keyword is used to declare exceptions. It specifies that there may occur an exception in the method. It doesn't throw an exception. It is always used with a method signature. 

## Java Exception Handling Example

Let's see an example of Java Exception Handling in which we are using try-catch-finally exception.

**JavaExceptionExample.java**

```
public class JavaExceptionExample{  
    public static void main(String args[]){  
        try{  
            //code that may raise exception  
            int data=100/0;  
        }catch(ArithmeticException e){System.out.println(e);}  
        //rest code of the program  
        System.out.println("rest of the code...");  
    }  
}
```

**Test it Now****Output:**

```
Exception in thread main java.lang.ArithmeticException:/ by zero  
rest of the code...
```

In the above example, 100/0 raises an ArithmeticException which is handled by a try-catch block.

---

## Common Scenarios of Java Exceptions

There are given some scenarios where unchecked exceptions may occur. They are as follows:

### 1) A scenario where ArithmeticException occurs

If we divide any number by zero, there occurs an ArithmeticException.

```
int a=50/0;//ArithmeticException
```



### 2) A scenario where NullPointerException occurs

If we have a null value in any variable, performing any operation on the vari

```
String s=null;  
System.out.println(s.length());//NullPointerException
```

### 3) A scenario where NumberFormatException occurs

If the formatting of any variable or number is mismatched, it may result into NumberFormatException. Suppose we have a **string** variable that has characters; converting this variable into digit will cause NumberFormatException.

```
String s="abc";  
int i=Integer.parseInt(s);//NumberFormatException
```

### 4) A scenario where ArrayIndexOutOfBoundsException occurs

When an array exceeds to it's size, the ArrayIndexOutOfBoundsException occurs. there may be other reasons to occur ArrayIndexOutOfBoundsException. Consider the following statements.

```
int a[]=new int[5];  
a[10]=50; //ArrayIndexOutOfBoundsException
```

## Java Exceptions Index

1. Java Try-Catch Block
2. Java Multiple Catch Block
3. Java Nested Try
4. Java Finally Block
5. Java Throw Keyword
6. Java Exception Propagation
7. Java Throws Keyword
8. Java Throw vs Throws
9. Java Final vs Finally vs Finalize
10. Java Exception Handling with Method Overriding
11. Java Custom Exceptions



 For Videos Join Our Youtube Channel: [Join Now](#)


Feedback


- Send your Feedback to [feedback@javatpoint.com](mailto:feedback@javatpoint.com)


Help Others, Please Share





Learn Latest Tutorials


 [Splunk tutorial](#)  
Splunk


 [SPSS tutorial](#)  
SPSS


 [Swagger tutorial](#)  
Swagger


 [T-SQL tutorial](#)  
Transact-SQL

 [Tumblr tutorial](#)  
Tumblr

 [React tutorial](#)  
ReactJS

 [Regex tutorial](#)  
Regex

 [Reinforcement learning tutorial](#)  
Reinforcement Learning

 [R Programming tutorial](#)  
R Programming







React Native tutorial  
React Native



Python Design Patterns  
Python Design Patterns



Python Pillow tutorial  
Python Pillow




Python Turtle tutorial  
Python Turtle




Keras tutorial  
Keras


Preparation




Aptitude  
Aptitude




Logical Reasoning  
Reasoning



Verbal Ability  
Verbal Ability




Interview Questions  
Interview Questions




Company Interview Questions  
Company Questions


Trending Technologies




Artificial Intelligence  
Artificial Intelligence




AWS Tutorial  
AWS




Selenium tutorial  
Selenium




Cloud Computing  
Cloud Computing




Hadoop tutorial  
Hadoop




ReactJS Tutorial  
ReactJS




Data Science Tutorial  
Data Science




Angular 7 Tutorial  
Angular 7




Blockchain Tutorial  
Blockchain



Git Tutorial  
Git




Machine Learning Tutorial  
Machine Learning




DevOps Tutorial  
DevOps


B.Tech / MCA




DBMS tutorial  
DBMS




Data Structures tutorial  
Data Structures




DAA tutorial  
DAA




Operating System  
Operating System




Computer Network tutorial  
Computer Network




Compiler Design tutorial  
Compiler Design



Computer Organization and Architecture  
Computer Organization



Discrete Mathematics Tutorial  
Discrete Mathematics



Ethics  
Ethic



Software  
Engineering  
Software  
Engineering



html tutorial  
Web Technology



Cyber Security  
tutorial  
Cyber Security



Automata  
Tutorial  
Automata



C Language  
tutorial  
C Programming



C++ tutorial  
C++



Java tutorial  
Java



.Net  
Framework  
tutorial  
.Net



Python tutorial  
Python



List of  
Programs  
Programs



Control  
Systems tutorial  
Control System



Data Mining  
Tutorial  
Data Mining



Data  
Warehouse  
Tutorial  
Data Warehouse

